CLAIMS

1. Method of verifying the integrity of a software application which is executable in a host terminal, characterised in that it comprises the following steps:

i) determining at least one series of control instructions forming an executable certificate (4, 15) for the software application, which can be executed by said host terminal during the execution of the software application to be verified (1, 11);

ii) on the host terminal, executing the software application to be verified (1, 11), receiving the executable certificate (4, 15) thus determined during step i), and executing the series of control instructions for said executable certificate in the memory context of said host terminal;

iii) comparing the result thus obtained through the execution of the control instructions with the result expected from an authentic software application; and.

iv) in the event of a positive comparison, continuing with the execution of the software application to be verified (1, 11).

2. Method according to claim 1, in which the host terminal is provided with a processor, characterised in that the series of control instructions forming the certificate (4, 15) is coded in a language which can be interpreted by said processor of the host terminal.

3. Method according to claim 1, in which the host terminal is provided with a virtual machine which is capable of emulating a processor, characterised in

that the series of control instructions forming the certificate (4, 15) is coded in a language which can be interpreted by the virtual machine of the host terminal.

4. Method according to one of claims 1 to 3, in which the executable certificate includes a portion of the processing necessary for the satisfactory operation of the authentic application.

5. Method according to one of claims 1 to 4, characterised in that, in step i), provision is made to establish, in a secure environment, a card with the memory context of the authentic software application during the course of execution, and to determine, from the values of this memory card, the series of control instructions intended to form the executable certificate (4, 15).

6. Method according to one of claims 1 to 5, characterised in that, in step ii), the executable certificate (4, 15) for the host terminal emanates from an electronic processing circuit which is physically separated from the host terminal.

7. Method according to one of claims 1 to 6, characterised in that, in step ii), the recovery of the execution values of the memory context is effected by reading the values at the addresses of the various portions of the memory of the host terminal, these portions containing the executable instructions and the data intrinsic to the application to be verified.

8. Method according to one of claims 1 to 7, characterised in that, in step iii), the result obtained by the execution of said series of control instructions (4, 15) produces a signature for the application to be verified, this signature being calculated by said series of control instructions (4, 15) which uses the values of

the memory context of the software application to be verified during the course of execution of the application.

9. Method according to one of the preceding claims, characterised in that the software application comprises instructions which permit said series of control instructions (4, 15) to be loaded and executed in its memory context card by substituting at least one address for executing an instruction of said software application by at least one instruction address of the series of instructions which form the certificate.

10. Method according to one of the preceding claims, characterised in that the series of control instructions (4, 15) is selected in such a manner that the state of the memory context of one software application after the execution of the series of control instructions is identical and/or without any modification to the state of the memory context of the software application prior to the execution of the series of control instructions.

11. Method according to any one of claims 1 to 10, characterised in that the series of instructions forming the certificate (4, 15) is transported into a stream of data necessary for the execution of the software application to be verified.

12. Method according to any one of claims 1 to 11, characterised in that the software application to be verified is wholly or partially encoded, the correct deciphering of the software application being achieved in the event of integrity of the software application to be verified.

13. Apparatus for verifying the integrity of a software application which is intended to be executed in a host terminal for accomplishing the method according to one of claims 1 to 12, characterised in that it comprises processing

means capable of determining at least one series of control instructions (4, 15) for the software application (1, 11), which can be executed by said host terminal during the execution of the software application, and which forms an executable certificate of said software application, executing means for executing the series of instructions forming the certificate (4, 15) on the host terminal during the execution of the software application, comparison means for comparing the result thus obtained through the execution of the control instructions with the result expected from an authentic application, and means which are capable, in the event of a positive comparison, of continuing with the execution of the software application to be verified (1, 11).

14. Apparatus according to claim 13, characterised in that it comprises a smart card or any other secure circuit which is capable of containing the series of control instructions forming the certificate (4, 15), in that the host terminal is provided with a reader for reading a smart card or with a means for communicating with the secure circuit, and in that the means for executing the software application are provided in order to pick-up, in the smart card or in the secure circuit, the series of instructions forming the certificate during the execution of the software application to be verified.

15. Apparatus according to claim 14, characterised in that the host terminal is capable of returning, to the smart card or to the secure circuit, the signature produced by the series of control instructions, and in that the smart card or the secure circuit additionally comprises a software application verifying means which is capable of validating or invalidating the authenticity of the software application to be verified in dependence on the result of the comparison between the signature produced by the series of control instructions and a value for the signature which is known and previously stored in the smart card or in the secure circuit.

16. Apparatus according to claim 15, characterised in that, in the event of a negative comparison, the smart card is capable of modifying the operation of the software application to be verified.

17. Apparatus according to claim 15 or claim 16, characterised in that, in the event of a non-transmission of the signature in conformity with predetermined conditions, the smart card is capable of modifying the operation of the software application to be verified.

18. Apparatus according to one of claims 13 to 17, characterised in that, in the event of a negative comparison, the apparatus additionally comprises means which are capable of preventing the operation of the software application in the host terminal.

19. Apparatus according to one of claims 13 to 18, characterised in that the host terminal belongs to the group formed by data processing apparatuses, digital television decoders, equipment for visualising multimedia contents, micro-computers, smart cards, personal organisers, game consoles, mobile telephones or the like.

20. Apparatus according to one of claims 13 to 19, characterised in that the processing means are capable of determining a plurality of executable certificates (4, 15) which differ from one another according to a selected rate and/or condition.

21. Apparatus according to one of claims 13 to 20, characterised in that the processing means are capable of determining a plurality of executable

certificates (14, 15) which differ from one another according to a selected rate and/or a selected condition.